RESEARCH PROPOSAL

Security Analysis of Linux Containers over Cloud Computing Infrastructure

For registration to the degree of Doctor of Philosophy

IN FACULTY OF SCIENCE

THE IIS UNIVERSITY, JAIPUR

Submitted by

Deepika Saxena
Enroll. No.: ICG/2017/24631

Under the Supervision of

Dr. Navneet Sharma
Associate Professor (CS & IT)

Department of CS & IT

August, 2018

TOPIC:

Security Analysis of Linux Containers over Cloud Computing Infrastructure

INTRODUCTION:

Cloud computing has changes the scenario of software development and deployment. With the help of these new technology applications, data, platform and the infrastructure are all stored on the cloud. This approach makes it possible to deliver a range of low cost applications, anywhere, anytime to any customer. It helps in reduce the hardware and software costs and provides a flexible and easy way to maintain environment for application developers.

Due to the advantages of cloud technology, small and large scale enterprises to secure their applications software, databases and infrastructure to the cloud environment. Despite of all the advantages, the potential adopters of cloud are still in a confusion whether secure to cloud or not, because cloud computing is still in an immature state. Because the cloud security could result in exposure of confidential internal data of the organization. Therefore before secure to the cloud we need to analyze and plan carefully, to ensure the compatibility issue as well as the integrity of the organizational data.

CLOUD COMPUTING:

According to the US National Institute of Standards and Technology (NIST)"Cloud Computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

Table 1: Key Characteristics of Cloud Computing

Characteristic	Description	References
On-demand service	Services or computing capacity made available as needed.	Mell and Grance(2011)
Ubiquitous access	Services can be accessible over the internet	Armbrust et al. (2009), Mell and Grance(2011)
High elasticity and scalability	Computing capabilities can be elastically provisioned, and dynamically scaled up or down with demand.	Mell and Grance (2011), Qian et al. (2009)
Pay-per-use	Users are charged for actual use of service instead of by subscription	Armbrust et al. (2009), Staten (2008)
Location independence	Users can access data and services without knowing its physical location.	Iyer and Henderson (2010)
No upfront commitment	Organizations can start small and increase resources as needed	Armbrust et al. (2009)

The NIST describes three cloud computing service models:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

Iyer et al. (2010) divided cloud computing vendors based on different levels of abstraction. Cloud computing is an emerging area in the IT industry. Number of companies can be seen as major cloud service provide. Table 2 lists some of the major cloud service providers with divided into the three NIST service models

Table 2: Cloud service models

Service	Vendors	Description
IaaS	Amazon	Elastic Compute Cloud (EC2) virtualized cloud infrastructure; Simple Storage Service (S3) cloud storage
	Vmware	Vblocks: helps business customers build out clouds based on Vblock packages.
	Rackspace	Rackspace Cloud: Cloud Sites, Cloud Servers, Cloud Files.
	Joyent	Joyent Accelerators (virtual servers)
	3Tera	AppLogic: provides infrastructure solution
PaaS	Google	A platform for developing and hosting web applications in Googlemanageddata centers; supports Python and Java.
	Vmware	vCloud: manages applications within private clouds or has them federated on-demand to partner-hosted public clouds.
	Salesforce.com	Force.com Platform (Custom Cloud 2, Development Platform)
	Microsoft	Azure: a Windows-as-a-service platform consisting of the operating system and developer services
	Joyent	OpenSolaris
	IBM	Blue Cloud computing platform: enhances software development and delivery capabilities, particularly in large companies
	NetSuite	SuitCloud: offers on-demand products, development tools, and Services
	3Tera	Cloudware: offers applications and storage.
SaaS	Google	SaaS: Web-based communication, collaboration, & security applications including Gmail, Google Calendar, Google Talk, and Google Docs
	Salesforce.com	CRM (Sales Cloud 2, Service Cloud 2)
	NetSuite	SuitCloud: offers on-demand products, development tools, and Services
Iaas—Infrastructure as a Service; Paas—Platform as a Service; SaaS—Software as a Service.		

The deployment models defined by the cloud are:

- Public cloud
- Private cloud
- Hybrid cloud
- Community cloud

Table 3: The deployment models of Cloud Computing

Type	**Description**	**Reference**
Public Cloud	The cloud service policy, value and costing are defined by a service provide and used by general public.	The NIST Definition of Cloud Computing (Mell and Grance 2011)
Community Cloud	The cloud infrastructure is shared by several organizations belonging to a single group or community. The infrastructure is hosted by third party of by the community.	
Private Cloud	The cloud model for a single organization and is maintained by organization or third parties	
Hybrid Cloud	The cloud infrastructure is a combination of two or more cloud models (private, community or public)	

CLOUD COMPUTING CHALLENGES:

- Security and Privacy
- Lack of Standards
- Continuously Evolving
- Compliance Concerns

CONTAINER

(Pahl, 2015), Containerisation is a technology to virtualised applications in a lightweight way that has resulted in a significant uptake in cloud applications management. How to orchestrate the construction and deployment of containers individually and in clusters has become a central problem.

The cloud uses virtualisation techniques to achieve elasticity of large-scale shared resources (Mell and Grance, 2011). Virtual machines (VMs) are typically the backbone at the infrastructure layer. Containerisation in contrast allows a lightweight virtualisation through the bespoke construction of containers as application packages from individual images (generally retrieved from an image repository) that consume less resources and time.

A container holds packaged self-contained, ready-to-deploy parts of applications and, if necessary, middleware and business logic (in binaries and libraries) to run the applications.
Tools like Docker are built around container engines where containers act as portable means to package applications. This results in the need to manage dependencies between containers in multi-tier applications. An orchestration plan can describe components, their dependencies and their lifecycle in a layered

CONTAINER ORCHESTRATION

Applications are typically made up of individually containerized components (often called micro services) that must be organized at the networking level in order for the application to run as intended. The process of organizing multiple containers in this manner is known as container orchestration.

In modern development, applications are no longer monolithic, but instead are composed of dozens or hundreds of loosely coupled, containerised components that need to work together to allow a given app to function as designed. Container orchestration refers to the process of organising the work of individual components and application layers.

CONTAINER ORCHESTRATON WORK

While platforms like Apache Mesos, Google Kubernetes, and Docker Swarm each have their own specific methodologies for container management, container orchestration engines all allow users to control when containers start and stop, group them into clusters, and coordinate all of the processes that compose an application. Container orchestration tools allow users to guide container deployment and automate updates, health monitoring, and failover procedures.

DOCKER:

Docker is a container management service. The keywords of Docker are **develop**, **ship** and **run** anywhere. The whole idea of Docker is for developers to easily develop applications, ship them into containers which can then be deployed anywhere.

The **docker project** offers higher-level tools, working together, which are built on top of some Linux kernel features

Docker achieves this by creating safe, LXC (i.e. Linux Containers) based environments for applications called docker containers. These containers are created using docker images, which can be built either by executing commands manually or automatically through Dockerfiles.

FEATURE OF DOCKER

- Docker has the ability to reduce the size of development by providing a smaller footprint of the operating system via containers.

- With containers, it becomes easier for teams across different units, such as development, QA and Operations to work seamlessly across applications.

- You can deploy Docker containers anywhere, on any physical and virtual machines and even on the cloud.

- Since Docker containers are pretty lightweight, they are very easily scalable.

COMPONENTS OF DOCKER

Docker has the following components

- **Docker for Mac** – It allows one to run Docker containers on the Mac OS.
- **Docker for Linux** - It allows one to run Docker containers on the Linux OS.
- **Docker for Windows** - It allows one to run Docker containers on the Windows OS.
- **Docker Engine** – It is used for building Docker images and creating Docker containers.
- **Docker Hub** – This is the registry which is used to host various Docker images.
- **Docker Compose** – This is used to define applications using multiple Docker containers.

DOCKER MANAGEMENT TOOLS:

Docker management's tools are:

1. KUBERNETES
2. SWARM

KUBERNETS

Docker is great for running containers in one host, and provides all required functionality for that purpose. But in today's distributed services environment, the real challenge is to manage resources and workloads across servers and complex infrastructures.

One such tool is Google's Kubernetes, a strange-sounding Greek word that means "governor" or "commander". As the word might suggest, Kubernetes undertakes the cumbersome task to orchestrate containers across many nodes, utilizing many useful features.

- A container orchestration tool
- Consists of servers, nodes
- Manages desired state of services
- Abstracts the concept of server to a pool of resources
- Networks span the whole swarm
- Provides common tools for updates, maintenance, etc

SECURE PARAMETERS IN MIGRATION

MIGRATION

Cloud migration is the process of moving data, applications or other business elements from an organization's onsite computers to the cloud, or moving them from one cloud environment to another.

Migration is a process of moving a container from one server to another server. Migration can facilitate in providing fault tolerance as container or VM can be migrated to another host if the system experience failures.

CURRENT STATE OF CONTAINER MIGRATION RESEARCH:

- Most of the research is targeted at supporting decision making in cloud migration process.
- The best practices or actual migration procedure are not so widely examined.
- What kind of application changes might be needed and why, these questions are still unanswered.
- The maturity of the research seems to be still in a growing state: more real life case studies and experience reports would increase trust in validity of research and benefits of cloud migration.
- Applicability of cloud migration in industrial context is still at growing state.
- There is still very less tool support to automate and facilitate cloud migration tasks. Ahmed M. Mahfouz.(2017)

REVIEW OF LITERATURE

Moving to the cloud presents the enterprise with a number of risks which include securing critical information like the protection of intellectual property, trade secrets, personally identifiable information that could fall into the wrong hands. Making sensitive information available on the internet requires a considerable investment in security controls and monitoring of access to the contents. In the cloud environment, the enterprise may have little or no visibility to storage and backup processes and little or no physical access to storage devices at the cloud computing provider. Moreover, because the data from multiple customers may be stored in a single repository, forensic inspection of the storage media and a proper understanding of the access and deletion will be a significant challenge.

Five Principles of Cloud Computing

Resource pooling: Cloud computing providers harness large economies of scale through resources pooling. They put together a vast network of servers and hard drives and apply the same set of configurations, protection and the works for them.

Virtualization: Users do not have to care about the physical states of their hardware nor worry about hardware compatibility.

Elasticity: Addition of more hard disk space or server bandwidth can be done with just a few clicks of the mouse on-demand. Geographical scalability is also available in cloud computing – one can choose to replicate data to several data centers around the world.

Automatic/easy resource deployment: The user only needs to choose the types and specifications of the resources he require and the cloud computing provider will configure and set them up automatically.

Metered billing:

Users are charged for only what they use.

Once the client hosts data to the cloud there should be some guarantee that access to that data will only be limited to the authorized access. Inappropriate access to customer sensitive data by cloud personnel is another risk that can pose potential threat to cloud data (Zhang and Zhou, 2009).

This paper reports our experiences and observations gained from the concept of Virtualization is a assemble of hardware and software engineering that makes Virtual Machines (VMs) and empowers antithetical active frameworks to protect continually on the related physical stage .Virtualization refers to technologies doomed to commit a layer of abstraction between computer hardware frameworks and the software running on them . Virtualization method to derive a virtual detail of a appliance or asset, for example, a server, storage device, settle or ultimately a working framework where the strategy partitions the asset into at least such execution environments

Although several dozens of cloud based services are being offered, the authors feels that the landscape is still fragmented and there are no comprehensive guidelines for designing, developing and deploying solutions to leveraging cloud computing. The observation is that there is hardly any guidance available for secure existing systems to cloud computing in terms of software engineering aspects

By Letaifa highlight the challenges that arise when porting legacy applications on cloud environments and propose enabling technologies for overcoming these challenges and the corresponding limitations. We emphasize on application analysis and aspects oriented to quality of service, multi-tenancy contexts that affect the application behavior at runtime, as well as characterization of providers in order to support decision making regarding the optimum ones to

host the aforementioned legacy applications following their analysis. Asma, (2010) presented two major types of virtualization architecture: facilitated and endangered metal. In facilitated architecture, a engaged framework (OS) is installed on the hardware first. At that relate the output called a hypervisor or virtual machine screen is confirmed. This produce is utilized to introduce antithetical visitor deal frameworks, or virtual machines, on the equipment. Applications are formerly confirmed and keep running on the virtual machines in the same manner as on a physical machine. With caught metal design, the hypervisor is confirmed straightforwardly on the equipment as facing on top of a fundamental engaged framework.

Challenges that may be encountered during the secure of an application on Cloud platforms

- Unknown internal structure
- Lack of knowledge for infrastructure environment
- Multi-tenancy influence
- Variable configuration based on user preferences
- Various application types with different characteristics and usage of the resources

The author (rashmi) defines the needs and benefits of cloud computing, but also identifies different types of challenges during cloud migration and divides them into three categories:

1. **Business Factors**
    a. Existing Investments in IT
    b. Costs
    c. Data Security
    d. Provisioning
2. **Technical Factors**
    a. Existing Infrastructure
    b. Security Architecture
    c. Complexity
    d. Network and Support
    e. IT Skills
    f. Service Level Agreements

3. Ease of Implementation

After extensive literature survey, author demanded a need for well-defined process for cloud migration and suggested a Five Phased Waterfall Model for cloud migration. The model is based on Iterative Waterfall Model from Software Development Life Cycle (SDLC). The waterfall model is modified with adding a feedback path at each stage. Therefore if a defect is detected any stage of migration, we can go back and improve.
The author finds the possibility of enhancement of the proposed model with the help of more case studies and research in this field.

Venkatraman, focuses on the lack of user-centric architecture, and identifies the traditional system design, implementation and testing issues prevalent in engineering the cloud services for container, in particular interoperability and extensibility issues with the SaaS. Further, the paper aims to propose a software engineering research roadmap to address these issues.
The authors identified a number of questions related to cloud services that play an important role in their adoption and still remain unanswered. Some of these unanswered questions are listed below:

- What are the cloud models offered in the market?
- What are the software engineering issues related to the technological and non-technological concerns present in the cloud agenda worldwide?
- How can each service provider's research perspectives and cloud agenda influence the secure to cloud services?
- How can each country's strategic cloud agenda influence the global cloud adoption?
- What are the research initiatives that could be undertaken by software engineering experts to address the issues present in the cloud agenda?
- What are the open software engineering research challenges in container to the cloud?
- How can we establish a software engineering research roadmap for the future of the cloud?

Zahir Tari, Xun Yi (2015) proposed that the Security of stored data, access management, data utilization management, and trust are among the primary security aspects in cloud computing. In cloud computing, data confidentiality and user authentication are correlated. The cloud is an open platform; it's susceptible to malicious attacks of continuously evolving natures. However, depending on the cloud deployment model, key management (assignment, distribution, and revocation) must be efficient and manageable at a large scale. A particularly promising approach to improving security in cloud computing is the use of cryptographic methods. However, in cloud deployment models, data and application control is delegated; hence traditional policy-based enforcement presents a number of challenges.

The paper identified the following research challenges in the field of cloud container:

- How to create and maintain a fine-grained holistic instance model of an enterprise's IT, which considers, besides the process, also the services and their realization?
- How to ensure manageability of this model and provide the abstraction and extraction methods required for migration of the services' realizations?
- How to enable the container of the service realizations and adapt these applications to use Cloud services?
- How to support a container model which takes into consideration all layers and components of composite enterprise applications discovering the enterprise IT model is an open issue

In simple terms 'cloud' refers to an IS functionality accessed by users through a thin client (web browser, mobile applications, etc.) while the software, hardware and data are stored on remote servers. Its popularity as an approach reflects claims that such 'shared resources' model will allow organization to access more advanced and flexible IS services at a lower aggregate cost. This paper argues that cloud migration requires organizations to consider a range of additional success factors, including how to accommodate new delivery modes (e.g. different pricing models) and manage new risks (e.g. data outside organizational firewalls, etc.). Despite burgeoning interest in the topic, specific empirical research and, more importantly, specific practitioner guidance remains limited. This paper therefore concludes by presenting a

preliminary conceptual model (informed by Systems Thinking) along with steps to operationalise it and suggests empirical options to test this model.

Jamshidi et al. conducted a systematic literature review (SLR) on legacy software cloud migration research in 2013. The review looks into 23 research papers published from 2010 to 2013. The objective of the paper is to systematically identify and taxonomically classify available evidence on cloud migration and provide a complete comparison report to analyze potential and limitations of existing research. The paper provides, focus on proposed methods, techniques, and best practices and techniques used in legacy-to-cloud migration.

The paper defines a cloud migration reference model – Cloud RMM.

- Planning: feasibility study, requirements analysis, decisions of providers and services, migration strategies.
- Execution: code modification, architecture extraction, data extraction, and transformation.
- Evaluation: deployment, testing, validation.

Research Trend and Future Directions identified are:

- Needs for a comprehensive migration framework
- Needs for architectural adaptation support
- Needs for self-adaptive cloud-enabled systems
- Solutions to address crosscutting concerns
- Automated support in migration process

The main objective of the research is to identify common migration processes and divide them into operational level activities. Prime research motive is to develop a cloud migration system, with minimum complexities. The purpose of the research is to investigate the difference between 3 deployment models IaaS, PaaS and SaaS, during cloud migration. The motivation behind the research is to provide a transparent, agile and generic migration process, which suits for variety of business models.

The research work tries to answer the following questions of organizations who wish to migrate to cloud :

- How to move and migrate to the cloud
- What are the common migration processes at each layer of cloud computing (IaaS, SaaS and PaaS)

The paper identifies the following challenges:

- Immaturity in terms of established procedure for cloud migration.
- Not adequate tools to support the migration process.

Alexis WallskogPappas The thesis work presents the state-of-the-art on migration to the cloud from an on-site legacy software application. According to the author there is no solution that fit all types of legacy software applications migration. The organization has to take in consideration what kind of application they have and what result they aim to achieve with the migration. Decisions made by the organization will determine the choice of methodology and how the migration will be executed. What is certain is that no matter what tools or methodologies, an organization choose the migration will require a lot of time, planning and manual work.

The research work concludes that

- In the current situation there are no methods and tools covering most of the aspects on how to migrate a legacy application to the cloud.
- Cloud Migration research area new and complex. Because there are more parties involved
- More case studies are required.
- More tools and methods need to be developed.

Legacy system migration to the cloud brings both great challenges and benefits, so there exist various academic research and industrial applications on legacy system migration to the cloud. By analyzing the research achievements and application status, the authors divide the existing migration methods into three strategies according to the cloud service models integrally. Different processes need to be considered for different migration strategies, and different tasks

will be involved accordingly. The similarities and differences between the migration strategies are discussed, and the challenges and future work about legacy system migration to the cloud are proposed. The aim of this paper is to provide an overall presentation for legacy system migration to the cloud and identify important challenges and future research directions.

Migration to SaaS requires to consider the specific migration strategy according to legacy system and existing SaaS. If existing SaaS has the same business functionality of legacy system, users can replace legacy system by SaaS. When some business functionality has been realized by existing SaaS, legacy system can be modernized by revising legacy system based on existing SaaS

The paper (rashmi) identifies categories and compares the existing research work in the area of legacy to cloud migration. It reviews thirty selected paper from 2009 till 2014. The primary objective of the paper is to identify, taxonomically classify and systematically compare existing research, focused on planning, executing and validating legacy to cloud migration. The paper also answers the following questions:

- What are the motivations behind migration to the cloud?
- What are the existing tasks, methods, and techniques to support secure migration of legacy on-premises software to cloud? In addition, what all tool support is available to achieve the objectives?
- What are the existing research themes? What should form future research dimensions in legacy to- cloud migration?

The result of SLR was identification of research challenges and future direction in the field:

- There is a need of more real time case studies from industry and academia in cloud migration field.
- The author proposed a five phased model for cloud migration but identifies the need of more frameworks.
- More solutions are required for Security concern in cloud.

MOTIVATION TO THE WORK:

- Cloud computing is gaining popularity in IT industry, due to its flexibility, scalability and attractive *pay-per-use* business model.
- Much of research in cloud computing has concentrated on technical issues such as designing, deployment, performance and interface requirements. The research within the organizational studies is rare.
- Most of the companies are facing challenges during adaptation of cloud-based solutions, especially in terms of migrating an existing legacy application to the cloud.
- It is difficult to discover the complex dependencies of legacy applications on the underlying IT environment.
- Lack of a systematic process to guide project managers and application developers in making informed cloud selection and migration decisions
- There are no comprehensive guidelines for designing, developing and deploying solutions for migrating software systems to cloud computing.
- **As per the observation of review all papers directly or indirectly identify the need for more case studies in Live migration field. The case studies which cloud help the cloud provider to identify the changes to be done during migrating a legacy system to cloud.**

OBJECTIVES OF RESEARCH:

- To provide the flexible and secure Infrastructure using dockers and containers for application hosting.
- To Analyze the Failure of Host using own algorithm for Automatic container
- To establish a layer between the user & amp; cloud which will act as controller node as Swarm or Kubernetes machines in Auto scaling Environment.
- Python based UI to maintain and monitoring Cloud based orchestrization.

METHODOLOGY

In order to achieve the above mentioned objectives the following methodology will be followed:

- **Phase 1:** To provide automatically scalability environment for Container services.
- **Phase 2:** -Maintain the secure orchestration of lXC by using swarm or kubernetes.
- **Phase 3:** Provide the environment for hardware level resource provisioning over orchestration.
- **Phase 4:** Planning for integration for cloud VM to container for automatically container and VM switching using customized designed python based Monitoring tool. To analyze containers load on more traffic over cloud for HA(high availability)

TOOLS & TECHNIQUES

Open source automated AWS tools will be used to create cloud, Python, Swarm, graphana and kubernetes, docker are some other tools.

Other tools needed as per requirement of the proposed research, may be used from time to time.

RESEARCH APPROACH

- In ***phase I*** objective 1 and 2 will be answered by performing extensive literature survey where relevant theory and frameworks regarding Legacy to Cloud Migration.
- *In* ***phase II*** Surveys will be performed in different organizations, which are and have been involved in legacy migration projects. Changes needed during migration of a legacy application to SaaS Cloud, will be identified by conducting interviews and case studies. Survey participants could be industry and academia personnel's.
- In ***phase III*** the results of the second phase would be investigated. With the help of few statistical tools, the analysis on the collected data would be done. With the help of analyzed results the evaluation of the theoretical model would be performed.

WORK PLAN

Phase Number	Approximate time duration	Phase description
1	For the first 5-6 Month	Literature Survey
2	For the first 11-12 Month	Case studies
3	For the next 5-6 Months	Data Analysis/Evaluation
4	For the next 6-8 Months	Performance analysis of proposed data model
5	At least 5-6 Months	Thesis Design.

PLACE OF WORK AND FACILITIES AVAILABLE

The research work will be undertaken with the help of online tools and cloud applications. The Case Study and Interviews would be conducted on personal basis or through social network sites like LinkedIn, Facebook etc.

LIMITATIONS & ALTERNATIVE PLAN OF THE STUDY

Some tools and consultation may be required for the research work. With the help of university resources, workshops and online learning these limitations could be overcome.

REFERENCES

1. Ali Baber, M. and, M.A. Chauhan (2011) A tale of Migration to cloud computing for sharing experiences and observations. In Proceeding of the 2nd international workshop on Software Engineering for Cloud Computing, ACM, 50-56

2. Alexis WallskogPappas(2014) Migration of Legacy Applications to the Cloud: A Review on Methodology and Tools for Migration to the Cloud , 22, Umea University.

3. Ambrust, M., A. Fox, R.Griffith, A.D. Joseph, R. Katx, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica and M. Zaharia (2009) A view of Cloud Computing, ACM, 4, 50-58

4. Andrikopoulos,V., T. Binz , F. Leymann , S. Strauch (2013) How to adapt applications for the cloud environment: challenges and solutions in migrating applications to the cloud computing, Springer, 95, 493–535

5. Buyya, R., C.S.Yeo, S.Venugopal ,J.Broberg and I.Brandic (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the fifth utility. FuturGenerComputSyst, 25,599–616

6. Cloud Computing Use Cases group: Moving to the cloud. http://cloudusecases.org/Moving to the Cloud.pdf (2011)

7. Iyer, B., and J.C.Henderson (2010) Preparing for the Future: Understanding the Seven Capabilities of Cloud Computing, MIS Quarterly Executive 9, 117-131.

8. Jamshidi, P., A. Ahmad and C. Phel (2013) Cloud Migration Research : A Systematic Review", IEEE Transactions on Cloud Computing , pp.142-157
9. Khadka, R., A.Seidi , A. Idu, J. Hage and S. Jansen(2013) Legacy to SOA Evolution: A Systematic Literature Review. In Migrating Legacy Applications: Challenges in Service Oriented Architecture and Cloud Computing Environments. pp.40-70, IGI Global

10. Khajeh-Hosseini A, D.Greenwood andI.Sommerville (2010) Cloud migration: a case study of migrating an enterprise IT system to IaaS. In: Proceedings of the 3rd IEEE international conference on cloud computing. pp 450–457

11. Mell, P. and T. Grance (2011) The NIST definition of cloud computing (draft), NIST special publication (800), p 145

12. Mohan, T.S. (2010) Migrating into a Cloud In Cloud Computing: Principles and Paradigm. pp. 43-56, Willey

13. NIST, NIST Definition of cloud computing v15, NIST, Editor. 2009, National Institute of Standards and Technology: Gaithersburg, MD (2009).

14. Pahl, C. , X. Huanhuan and R. Walshe (2013)A Comparison of On-Premise to Cloud Migration Approaches, Springer, 8, 212-226

15. Rashmi, S. Mehfuz and G. Sahoo(2012) A Five Phased Approach for the Cloud Migration, International Journal of Emerging Technology and Advanced Engineering, 2, 286-291

16. Zhao JF, JT Zhou (2014) Strategies and Methods for Cloud Migration, International Journal of Automation and Computing, 11, 143-152

17. By https://www.hpe.com/in/en/what-is/container-orchestration.html
18. Liang-Jie Zhang and Qun Zhou- IBM T.J. Watson Research Center, New York, USA, Cloud Computing Open Architecture, 2009 IEEE International Conference on Web Services
19. Tasneem Salah, M. Jamal Zemerly(2017), Performance Comparison between Container-based and VM-based Services , International Journal of Information Sciences, January 2017, Dubai, UAE, Volume 131, Issue 1-4,

20. Claus Pahl_ and Antonio Brogi,(2017), Cloud Container Technologies: a State-of-the-Art Review Carnegie Mellon University, Pittsburgh, USA

21. Major Hayden.,(2015), Securing Linux Containers, SANS Institute InfoSec Reading Room, USA

22. Cloud Computing and Sustainability: The Environmental Benefits of moving to the Cloud. Technical Report, Accenture, USA, 2010, 1 – 16.

23. Li, H., Sedayao, J., Hahn-Steichen, J., Jimison, Ed., Spence, C. and Chahal, S., Developing an Enterprise Cloud Computing Strategy. Technical Report, Intel Corporation, USA, 2009, 1 – 16.

24. Hitesh A Bheda, Chirag S Thaker "Virtualization Driven Mashup Container in Cloud Computing PaaS Model "Proceedings published by International Journal of Computer Applications® (IJCA) International Conference on Computer Communication and Networks" CSI- COMNET-2011 pp. 9-14.